11

An external command is
launched to capture a
system command
environment

12

The system command
environment is stored in
an array of strings

13

At least one parameter
string is added to the
array of strings

14

The command is linked
to the command
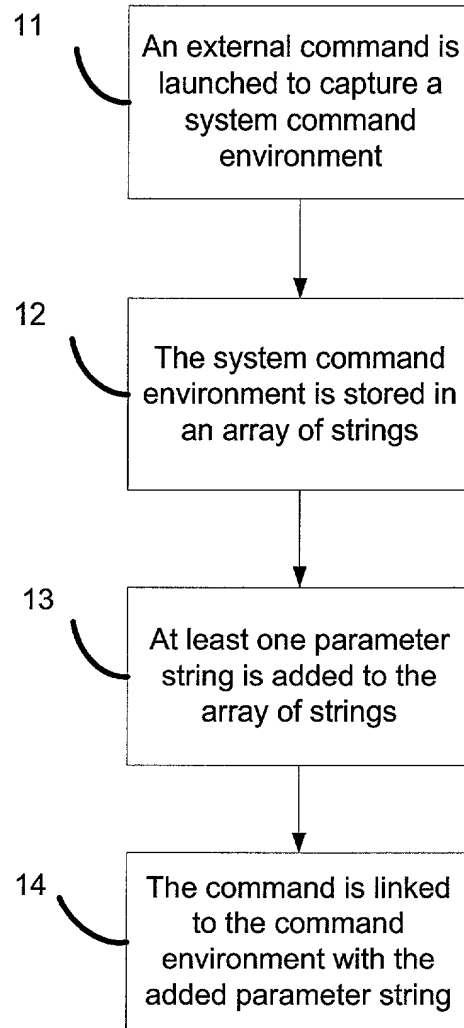environment with the
added parameter string

# FIG. 1

10

# 2/4

```
public int runCommand(String stepName, String cmdName, String args[]) {
    String cmd = prefs.getProperty(cmdName + ".cmd");
    Vector envp = new Vector();
    String vxargPrefix = "reportpar" + reportIndex + ".";
25
    envp.addElement("VXARG0="+ prefs.getProperty(vxargPrefix + "0", "novalue"));
    envp.addElement("VXARG1=" + prefs.getProperty(vxargPrefix +"1", "novalue"));

    envp.addElement("IWDA_WHOAMI=" + prefs.getProperty("env.whoami", "novalue"));
30  envp.addElement("IWDA_PATH=" + prefs.getProperty("env.iwdpath", "novalue"));

    envp=loadEnvp(envp);

    int rc = 0;
35
    for(int i=0; i<args.length; i++) {
      if(args[i]==null) break;
        if(platform.startsWith("AIX")){
        cmd = cmd+ " "+ args[i];
40        }
        else{
        cmd=cmd + " \"" + args[i] + "\"";
        }
    }
45    try {

    String envp2[]= new String[envp.size()];
      envp.copyInto(envp2);
      Process cmdProc = Runtime.getRuntime().exec(cmd,envp2);
50    rc=cmdProc.exitValue();
      if(rc!=0) throw new Exception("rc="+rc);
      }
    catch (Exception e) {
      return rc;
55    }
    return rc;
  }
```
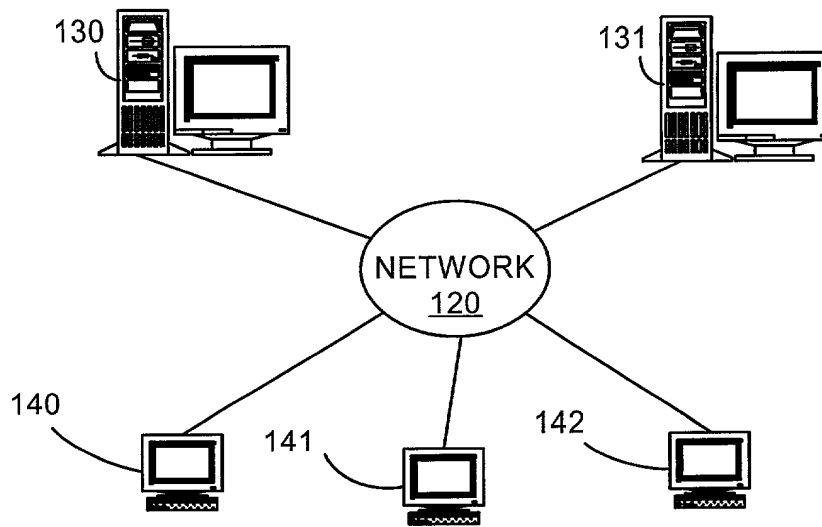
# FIG. 2

**20**

```
       public Vector loadEnvp(Vector envp) {
       String FS = File.separator;
       String envpFile=prefs.getProperty("env.iwddata") + FS + "tmp"+FS +"envpfile";
       String cmd;
65
         if(platform.startsWith("AIX")) {
            cmd=prefs.getProperty("env.iwddata") + FS + "tmp" + FS + "createEnv.sh";
          try {
            FileOutputStream cmdos = new FileOutputStream(cmd);
70          PrintWriter cmdLog = new PrintWriter(cmdos);
            cmdLog.println("#!/bin/sh");
            cmdLog.println("env > "+envpFile);
            cmdLog.close();
            Process cmdProc2 = Runtime.getRuntime().exec("chmod 755 "+ cmd);
75          cmdProc2.waitFor();
            }catch (Exception io){}
            }
         else {
            cmd="db2cmd -c -i -w set";
80          cmd=cmd + " >" + envpFile;
            }

         try {

85          Process envProc =Runtime.getRuntime().exec(cmd);
            envProc.waitFor();
            FileReader    fin = new FileReader(envpFile);
            BufferedReader bin = new BufferedReader(fin);
            String envpLine;
90
           while((envpLine=bin.readLine()) != null) {
              envp.addElement(envpLine);
              }

95          bin.close();
           fin.close();
             }
           catch (Exception e) {
             logTrace(e);
100          }
          return envp;
        }
```

# FIG. 3

<u>60</u>

130    131

140

141    142

NETWORK
120

110

FIG. 4